

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-319569

(43)公開日 平成9年(1997)12月12日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	5 4 0		G 0 6 F 9/06	5 4 0 N
9/38	3 1 0		9/38	3 1 0 A

審査請求 未請求 請求項の数4 OL (全 15 頁)

(21)出願番号 特願平8-139138

(22)出願日 平成8年(1996)5月31日

(71)出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72)発明者 三尾 雅夫

東京都千代田区丸の内二丁目2番3号 三  
菱電機株式会社内

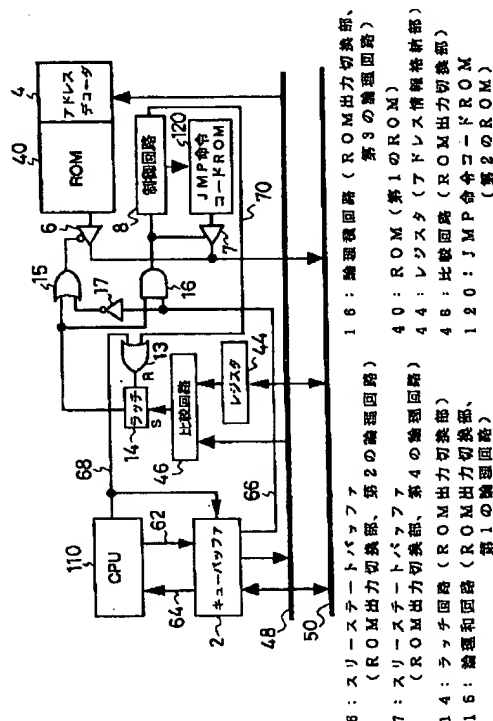
(74)代理人 弁理士 田澤 博昭 (外2名)

(54)【発明の名称】 マイクロコンピュータ

(57) 【要約】

【課題】 命令を先取りするキューバッファが設けられていると、ROM内のプログラムに生じた不具合を解消する機構がうまく動かない。

【解決手段】 比較回路46は、レジスタ44に格納されているアドレス情報とアドレスバス48上のアドレスデータとを比較する。ラッチ回路14は、比較回路46からの一致信号でセット状態になり分岐発生信号68またはJMP命令コード終了信号70によってリセット状態になる。リセット状態で、JMP命令コードROM120からデータバス50へのデータ出力は禁止される。



## 1

## 【特許請求の範囲】

【請求項1】 第1のROMに固定されたプログラムの各命令をCPUの実行に先立って先取りするとともに前記CPUが分岐命令を実行したことを示す分岐発生信号によってクリアされるキューバッファと、前記第1のROMのアドレス空間以外のアドレス空間に分岐する命令を固定した第2のROMと、アドレス情報を格納するアドレス情報格納部とを備えたマイクロコンピュータにおいて、前記アドレス情報格納部に格納されているアドレス情報と前記キューバッファのプログラムカウンタ値とを比較してそれらが一致したときに前記第2のROMからのデータ出力を許可するとともに前記第1のROMからのデータ出力を禁止し、前記分岐発生信号または前記第2のROMが命令を出力したことを示す命令コード終了信号を入力して前記第1のROMからのデータ出力を許可するとともに前記第2のROMからのデータ出力を禁止するROM出力切換部を備えたことを特徴とするマイクロコンピュータ。

【請求項2】 ROM出力切換部は、アドレス情報格納部に格納されているアドレス情報とアドレスバス上のアドレスデータとを比較する比較回路と、前記比較回路からの一致信号でセット状態になり、分岐発生信号または命令コード終了信号によってリセット状態になるラッチ回路とを備えたことを特徴とする請求項1記載のマイクロコンピュータ。

【請求項3】 ROM出力切換部は、ラッチ回路がリセット状態になるとキューバッファからのROMリード信号を通過させる第1の論理回路と、第1のROMの出力を入力し前記第1の論理回路の出力によって入力信号の通過状態が制御される第2の論理回路と、前記ラッチ回路がセット状態になると前記キューバッファからのROMリード信号を通過させる第3の論理回路と、第2のROMの出力を入力し前記第3の論理回路の出力によって入力信号の通過状態が制御される第4の論理回路とをさらに備えたことを特徴とする請求項2記載のマイクロコンピュータ。

【請求項4】 第1のROMに固定されたプログラムの各命令をCPUの実行に先立って先取りするとともに前記CPUが分岐命令を実行したことを示す分岐発生信号によってクリアされるキューバッファと、前記第1のROMのアドレス空間以外のアドレス空間に分岐する命令を固定した第2のROMと、アドレス情報を格納するアドレス情報格納部と、前記アドレス情報格納部に格納されているアドレス情報とCPUのプログラムカウンタ値とを比較する比較回路とを備えたマイクロコンピュータにおいて、前記比較回路が前記アドレス情報と前記CPUのプログラムカウンタ値との一致を示す一致信号を出力するとセット状態になり、第2のROMが命令を出力したことを示す命令コード終了信号によってリセット状態になるラッチ回路と、前記ラッチ回路がセット状態で

## 2

あるときに前記キューバッファから前記CPUへの命令の入力を禁止するとともに前記第2のROMから前記CPUへの命令の入力を許可するスイッチ回路とを備えたことを特徴とするマイクロコンピュータ。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 この発明は、プログラムを格納するROMを備えROM内のデータの先取り処理を行うマイクロコンピュータであって、ROM内のプログラムに生じた不具合を解消することができるマイクロコンピュータに関するものである。

## 【0002】

【従来の技術】 図8は従来のROM内のプログラムに生じた不具合を解消することができるシングルチップマイクロコンピュータを外部機器とともに示すブロック図である。図において、100はシングルチップマイクロコンピュータ（以下、単にマイクロコンピュータともいう）、200は修正プログラムが設定されたEEPROMである。マイクロコンピュータ100において、40はプログラムが固定されたROM、42は書き込み可能なメモリであるRAM、44は特定アドレスデータが設定されるレジスタ、46はアドレスバス48上のデータとレジスタ44の内容とを比較する比較回路、50はデータバス、52はシリアルデータ出力のためのシリアル出力端子、54はシリアルデータ入力のためのシリアル入力端子、110はCPU、120は分岐命令（JMP命令）が格納されているJMP命令コードROMである。なお、CPU110とシリアル出力端子52およびシリアル入力端子54との間にはシリアル入出力回路が設けられているが、図8では省略されている。

【0003】 次に動作について説明する。このマイクロコンピュータ100におけるROM40は、製造時に内容が固定され、以後、書き換え不可能ないわゆるマスクROMである。従って、一旦製造されたら、その内容を変更することはできない。しかし、現実には、ROM40内に固定されたプログラムの不具合が製造後に発見されることがある。

【0004】 マイクロコンピュータ100を製造した後にROM40内のプログラムにおいて不具合が発見された状況に対処するための手法として、例えば、特開昭63-156231号公報や特開平1-232447号公報に示された手法がある。図9は、その手法を説明するための説明図である。図9において、(a)は不具合を含むプログラムを示し、(b)は不具合が修正されたプログラムの流れを示す。その手法によると、マイクロコンピュータ100内に、レジスタ44が用意される。レジスタ44には、例えば、シリアル入力端子54を介して入力されたデータが設定される。JMP命令コードROM120には、JMP命令コード(jmp1~jmp4)が設定される。なお、jmp1~jmp4は、JM

## 3

P命令を構成する4バイトのデータを示す。JMP命令のオペランドは、RAM42内の所定の領域の先頭アドレスに対応した値を示す。

【0005】ROM40内のプログラムの不具合が発見された場合に、レジスタ44に、不具合部分の先頭アドレスが設定される。図9(a)において、不具合部分の先頭の番地に設定されている命令がop1～op4で示されている。そして、EEPROM200内に書き込まれた正しいプログラムが、シリアル出力端子52を介したデータによるアドレッシングによって、シリアル入力端子54を介して順次マイクロコンピュータ100に取り込まれる。そして、RAM42の所定の領域に設定される。所定の領域の先頭アドレスは、JMP命令コードROM120内のJMP命令のオペランドが指すアドレスである。このようにして、外部バスを持たないマイクロコンピュータ100においても、正しいプログラムが取り込まれる。なお、正しいプログラムを有する媒体はEEPROM200に限られない。他の媒体を用いることもできる。

【0006】その後、マイクロコンピュータ100は実稼働される。比較回路46は、マイクロコンピュータ100の実稼働時に、アドレスバス48に現れるアドレスデータとレジスタ44内の値とを比較する。それらが一致すると、比較回路46は、一致信号をJMP命令コードROM120に対して出力する。一致信号に応じて、JMP命令コードROM120内のJMP命令コードがデータバス50に出力される。このとき、ROM40からデータバス50へのデータ出力は禁止される。そして、CPU110は、データバス50からJMP命令コードを取り込む。そして、内部に有するプログラムカウンタの値をJMP命令のオペランドが示す値にする。従って、以後、RAM42に書き込まれた正しいプログラムが実行される。RAM42に書き込まれた正しいプログラムの実行後ROM40のプログラムに戻るために、RAM42内の正しいプログラムの最後の部分に、プログラムカウンタを変更するような分岐命令が格納される。

【0007】以上の処理によって、図9(a)に示すようにROM40内の不具合部分に記載されている命令による処理が行われる段階に達すると、不具合のあるプログラムに代わって、図9(b)に示すようにRAM42内の正しいプログラムが実行される。

【0008】マイクロコンピュータ100として、命令を先取りして一時格納し格納された命令をCPU110に供給するキューバッファを備えたものがある。そのようなマイクロコンピュータ100では、キューバッファが命令コードを構成するデータをROM40から入力する。そして、所定数を越える数のデータを保持すると、先取り処理を中断する。

【0009】比較回路46が一致信号を出力すると、キ

## 4

ューバッファがJMP命令コードROM120内のJMP命令コードを入力することになるが、そのとき、ROM40からデータバス50へのデータ出力は禁止される。すなわち、ROM40とデータバス50との接続は切り離される。JMP命令コードが複数バイトのデータで構成されている場合には、JMP命令コードを構成する複数データのうちの一部分のみがキューバッファに取り込まれた状態で、先取り処理が中断されることがある。すると、キューバッファはJMP命令コード中の残りのデータを欲しいためにJMP命令コードROM120とデータバス50との接続を維持することを要求しているが、CPU110は他の分岐命令を実行するためにROM40をデータバス50に接続することを要求するといった事態が生ずる。例えば、JMP命令コードを構成する複数データのうちの一部分のみがキューバッファに取り込まれた状態で、マイクロコンピュータ100が割込み処理を実行しなければならないといった事態がある。ソフトウェア割込みの場合には、ROM40に格納されている割込み命令が、JMP命令コードROM120内のJMP命令コードの読み込み以前にキューバッファに読み込まれていることになる。また、ハードウェアによる割込みの要求があると、CPU110が任意の命令を実行開始しようとする、いつでもその命令の実行に代えて割込み処理が実行される。このように、キューバッファを備えたマイクロコンピュータ100では、JMP命令コードROM120内のJMP命令コードの読み出し要求と割込み処理等が競合すると制御不能に陥る。

## 【0010】

【発明が解決しようとする課題】従来のマイクロコンピュータは以上のように構成されているので、命令を先取りするキューバッファが設けられていると、ROM内のプログラムに生じた不具合を解消する機構の制御が困難になるという課題があった。

【0011】この発明は上記のような課題を解決するためになされたもので、キューバッファを有する場合に、内蔵するROM内のプログラムの不具合に確実に対処することができるマイクロコンピュータを得ることを目的とする。

## 【0012】

【課題を解決するための手段】請求項1記載の発明に係るマイクロコンピュータは、アドレス情報格納部に格納されているアドレス情報とキューバッファのプログラムカウンタ値とが一致したときに、第1のROMのアドレス空間以外のアドレス空間に分岐する命令を固定した第2のROMからのデータ出力を許可するとともにプログラムが固定されている第1のROMからのデータ出力を禁止し、分岐発生信号または第2のROMが命令を出力したことを示す命令コード終了信号を入力して第1のROMからのデータ出力を許可するとともに第2のROMからのデータ出力を禁止するROM出力切換部を備えた

ものである。

【0013】請求項2記載の発明に係るマイクロコンピュータは、ROM出力切換部が、アドレス情報格納部に格納されているアドレス情報とアドレスバス上のアドレスデータとを比較する比較回路と、比較回路からの一致信号でセット状態になり分岐発生信号または命令コード終了信号によってリセット状態になるラッチ回路とを含むものである。

【0014】請求項3記載の発明に係るマイクロコンピュータは、ラッチ回路がリセット状態になるとROMリード信号を通過させる第1の論理回路と、第1のROMの出力を入力し第1の論理回路の出力によって入力信号の通過状態が制御される第2の論理回路と、ラッチ回路がセット状態になるとROMリード信号を通過させる第3の論理回路と、第2のROMの出力を入力し第3の論理回路の出力によって入力信号の通過状態が制御される第4の論理回路とをさらに含むものである。

【0015】請求項4記載の発明に係るマイクロコンピュータは、比較回路がアドレス情報格納部に格納されているアドレス情報とCPUのプログラムカウンタ値との一致を示す一致信号を出力するとセット状態になり、第2のROMが命令を出力したことを示す命令コード終了信号によってリセット状態になるラッチ回路と、ラッチ回路がセット状態であるときにキューバッファからCPUへの命令の入力を禁止するとともに第2のROMからCPUへの命令の入力を許可するスイッチ回路とを備えたものである。

【0016】

【発明の実施の形態】以下、この発明の実施の一形態を説明する。

実施の形態1。図1はこの発明の実施の形態1によるマイクロコンピュータの構成を示すブロック図である。ここでは、シングルチップマイクロコンピュータを例示する。図において、2は命令を先取りして一時格納するキューバッファ、4はアドレスバス48上のデータをデコードしアドレスデータに対応したROM（第1のROM）40の番地からデータを出力させるアドレスデコーダ、6はLレベルの制御信号によって信号通過状態になりROM40から出力されたデータをデータバス50に通過させるスリーステートバッファ、7はHレベルの制御信号によって信号通過状態になりJMP命令コードROM（第2のROM）120から出力されたデータをデータバス50に通過させるスリーステートバッファ、8はJMP命令コードROM120に格納されている命令コードの出力順序を制御する制御回路、13はCPU110からの分岐発生信号68と制御回路8からのJMP命令コード終了信号70との論理和をとる論理和回路（以下、OR回路と記す）、14は比較回路46からの一致信号でセット状態になりOR回路13からの信号でリセット状態になるラッチ回路、15はキューバッファ

2からのROMリード信号66がインバータ回路17で反転された信号とラッチ回路14の出力との論理和をとる論理和回路（以下、OR回路と記す）、16はラッチ回路14の出力とROMリード信号66との論理積を制御回路8およびスリーステートバッファ7に出力する論理積回路（以下、AND回路と記す）、44は特定アドレスデータが設定されるレジスタである。62はCPU110がキューバッファ2に対して出力するアドレスデータを示し、64はキューバッファ2からCPU110へ与えられるデータを示す。

【0017】なお、ROM出力切換部は、この場合には、比較回路46、ラッチ回路14、OR回路15、AND回路16、スリーステートバッファ6およびスリーステートバッファ7で実現される。また、アドレス情報格納部は、レジスタ44で実現される。第1～第4の論理回路は、それぞれ、OR回路15、スリーステートバッファ6、AND回路16、スリーステートバッファ7で実現される。

【0018】図2はマイクロコンピュータの動作を説明するためのタイミング図である。図において、(a)は、クロック信号を示す。(b)は、キューバッファ2内のデータ数を示す。(c)は、CPU110の動作を示す。括弧内は、キューバッファ2からCPU110に入力されたデータを示す。(d)は、CPU110がキューバッファ2に対して出力するアドレスデータ62を示す。CPU110が有するプログラムカウンタの値でもある。(e)は、データバス50からキューバッファ2に入るデータを示す。(f)は、ROMリード信号66を示す。(g)は、キューバッファ2がアドレスバス48に出力するアドレスデータを示す。キューバッファ2が有するプログラムカウンタの値でもある。(h)は、比較回路46から出力される一致信号を示す。(i)は、JMP命令コード終了信号70を示す。(j)は、ラッチ回路14の出力を示す。(k)は、分岐発生信号68示す。

【0019】次に動作について説明する。アドレスバス48へのアドレスデータの出力は、キューバッファ2によって行われる。キューバッファ2は、格納しているデータ数が「1」または「0」バイトになるとROM40から命令コードを読み出す。命令コードは、2バイト単位でROM40から読み出されるとする。なお、CPU110は、命令実行中にメモリアクセスが必要になると、直接、該当アドレスをアクセスする。

【0020】このマイクロコンピュータが製造されたときには、ROM40には所定のプログラムが固定されている。また、JMP命令コードROM120には、RAM（図示せず）内の特定領域の先頭番地に分岐するJMP命令が固定される。なお、JMP命令は、4バイトのデータで構成されているとする。特定領域とは、不具合プログラムに代わる正しいプログラムを設定するために

ユーザに開放されたRAM内の領域である。ROM40内のプログラムにおいて不具合が発見されると、ユーザは、例えば図8に示されたEEPROM200を用いた機構によって、レジスタ44に不具合プログラムの先頭番地を設定するとともに、RAM内の特定領域に正しいプログラムを書き込む。ここでは、レジスタ44には、特定アドレスデータとして「PC1+2」が設定されるとする。すなわち、「PC1+2」が不具合プログラムの先頭番地である。

【0021】以下、レジスタ44に不具合プログラムの先頭番地が設定され、RAM内の特定領域に正しいプログラムが書き込まれた状態でマイクロコンピュータが動作する場合について説明する。図2に示すサイクル#1に入る前では、キューバッファ2は初期状態であったとする。すなわち、キューバッファ2内のデータ数は「0」であり、キューバッファ2のプログラムカウンタはアドレスデータ62によって設定される状態である。そして、CPU110は、キューバッファ2に対してアドレス「PC1」のデータを要求したとする。すると、キューバッファ2は「PC1」をプログラムカウンタに設定する。そして、「PC1」をアドレスバス48に出力するとともに、ROMリード信号66を有意なレベルであるHレベルにする。ラッチ回路14の初期出力状態はリセット状態であるとする。すなわち、ラッチ回路14の出力はLレベルであるとする。すると、OR回路15の入力はともにLレベルであるから、スリーステートバッファ6が信号通過状態になる。アドレスデコーダ4は、アドレスバス48上の「PC1」をデコードして、該当番地に格納されている命令コードをROM40に出力させる。スリーステートバッファ6が信号通過状態にあるので、命令コードはデータバス50に出力される。キューバッファ2は、データバス50から命令コードを入力する。命令コードは2バイト単位で読み出されるので、ROM40はアドレス「PC1, PC1+1」に格納されている2バイトのデータop1, op2を出力し、キューバッファ2は、データバス50を介してデータop1, op2を入力する。すなわち、サイクル#1において、キューバッファ2内のデータ数は「2」になる(図2(b)参照)。

【0022】CPU110からアドレス「PC1」のデータの要求があったので、キューバッファ2は、データop1をCPU110に対して出力する。よって、キューバッファ2内のデータ数は「1」になる(図2(b)のサイクル#2参照)。データ数が「1」になったので、キューバッファ2は、命令の先取り処理を実行する。すなわち、「PC1+2」をアドレスバス48に出力するとともに、ROMリード信号66を有意なレベルであるHレベルにする。一方、データop1が1つの命令コードのオペコードであることをCPU110が認識すると(図2(c)のサイクル#2参照)、キューバッ

ファ2による命令の先取り処理の間、CPU110は、オペランドが格納されているアドレス「PC1+1」のデータをキューバッファ2に要求する(図2(d)のサイクル#2~#4参照)。キューバッファ2は、その要求に応じて、保持しているデータop2をCPU110に出力する。よって、キューバッファ2内のデータ数は「0」になる(図2(b)のサイクル#3参照)。

【0023】CPU110は、サイクル#5~#8において、データop1, op2からなる命令に応じた処理を行う。図2(c)では、アドレス「PC2」からデータを読み込んで加工を行い、結果をアドレス「PC2」に書き込む処理が例示されている。この間、CPU110は直接にアドレス「PC2」をアクセスする(図2(d)のサイクル#4~#8参照)。なお、図2(e)における斜線は、データバス50が命令コードの転送以外に用いられている期間を示す。

【0024】レジスタ44には不具合プログラムの先頭番地として「PC1+2」が設定されているので、キューバッファ2が「PC1+2」をアドレスバス48に出力すると、比較回路46は、一致信号をHレベルにする(図2(h)参照)。すると、ラッチ回路14の出力がHレベルになる(図2(j)参照)。ラッチ回路14のHレベルの出力によってOR回路15の出力がHレベルになるので、スリーステートバッファ6が閉じる。従って、ROM40からのデータはデータバス50に出力されない。ラッチ回路14の出力はAND回路16の一方の入力となるので、AND回路16はROMリード信号66を通過させうる状態になる。キューバッファ2はROMリード信号66をHレベルにしているため、AND回路16の出力はHレベルになる。よって、スリーステートバッファ7は信号通過状態になる。制御回路8は、AND回路16の出力がHレベルになると起動し、JMP命令コードROM120から、JMP命令を順序立てて出力させる。

【0025】JMP命令コードROM120から、まず、命令コードの最初の2バイトであるデータjmp1, jmp2がクロック信号に同期して順次出力される。データjmp1, jmp2は、データバス50を介してキューバッファ2に入力する(図2(e)のサイクル#3, #4参照)。キューバッファ2内のデータ数が「2」になる(図2(b)のサイクル#4参照)、キューバッファ2は、命令の先取り処理を中断する。

【0026】CPU110は、データop1, op2で構成される命令コードの処理が終了すると、キューバッファ2に、アドレス「PC1+1」の次のアドレス「PC1+2」の命令コードを要求する。その要求に応じて、キューバッファ2は、保持しているデータjmp1をCPU110に出力する。この結果、キューバッファ2内のデータ数が「1」になる(図2(b)のサイクル#5参照)。キューバッファ2は、命令の先取り処理を再開する。すなわち、キューバッ

2は「PCI+4」をアドレスバス48に出力するとともに、ROMリード信号66をHレベルにする(図2

(f), (g)のサイクル#9~#11参照)。ROMリード信号66に応じてJMP命令コードROM120からJMP命令の後半の2バイトのデータjmp3, jmp4が出力される。これらのデータjmp3, jmp4は、データバス50を介してキューバッファ2に入力する(図2(e)のサイクル#10, #11参照)。

【0027】一方、CPU110は、データjmp1がJMP命令のオペコードであったことを知り、JMP命令のオペランドを得るために、キューバッファ2に「PCI+3」のデータを要求する(図2(d)のサイクル#9~#11)。キューバッファ2は、その要求に応じて保持しているデータjmp2, jmp3をCPU110に転送する(図2(c)のサイクル#10, #11参照)。従って、キューバッファ2内のデータ数は「1」になる。なお、その前に、キューバッファ2内のデータ数は、データjmp3, jmp4を入力したときに「2」増えて「3」になっていた(図2(b)のサイクル#10参照)。JMP命令は4バイトで構成されているので、CPU110は、残りの1バイトを要求するために、アドレスデータ62として「PCI+5」を出力する。キューバッファ2は、この要求に応じて、保持しているデータjmp4をCPU110に転送する。従って、キューバッファ2内のデータ数は「0」になる。

【0028】制御回路8は、JMP命令コードがJMP命令コードROM120から出力され終わった時点で、JMP命令コード終了信号70をHレベルに立ち上げる(図2(i)参照)。ラッチ回路14は、OR回路13を介して入力されたJMP命令コード終了信号70の立ち上がりによってリセット状態になる(図2(j)参照)。すなわち、その出力はLレベルになる。従って、OR回路15はROMリード信号66を通過させる状態になるので、以後、ROMリード信号66がHレベルになると、ROM40からのデータが、データバス50を介してキューバッファ2に入力する。ラッチ回路14の出力を導入するAND回路16の一方の入力がLレベルになるので、AND回路16はLレベルになる。よって、スリーステートバッファ7は信号通過禁止状態になる。すなわち、以後、JMP命令コードROM120のデータは、データバス50に出力されない。

【0029】キューバッファ2は、サイクル#11において、命令コードを先取りするために「PCI+6」をアドレスバス48に出力するとともにROMリード信号66をHレベルにする(図2(f), (g)のサイクル#11~#13参照)。この時点では既にスリーステートバッファ6が信号通過状態になっているので、HレベルのROMリード信号66に応じて、ROM40から、アドレス「PCI+6, PCI+7」のデータ「op7, op8」がデータバス50に出力される。キューバッファ2

は、データバス50からデータ「op7, op8」を取り込む。

【0030】しかし、CPU110から、JMP命令の実行完了時に分岐発生信号68が出力される(図2

(k)参照)。キューバッファ2は、分岐発生信号68によってクリアされる。従って、データ「op7, op8」はキューバッファ2において消滅する。よって、キューバッファ2内のデータ数は「0」になる。また、キューバッファ2のプログラムカウンタは、次にCPU110からデータを要求されたときに、アドレスデータ62が示す値に設定される。なお、CPU110のプログラムカウンタは、JMP命令コードのオペランドが示す値に更新される。

【0031】以上のようにして、正しいプログラムに分岐するためのJMP命令コードをJMP命令コードROM120からデータバス50に出力させたいときのみ、JMP命令コードROM120からのデータを入力とするスリーステートバッファ7が信号通過状態になり、スリーステートバッファ6が信号通過禁止状態になる。そして、CPU110が正しいプログラムに分岐するためのJMP命令(jmp1~jmp4)を実行したら、ROM40からのデータを入力とするスリーステートバッファ6が信号通過状態になり、スリーステートバッファ7が信号通過禁止状態になる。以後、RAMに設定された正しいプログラムが実行されるが、そのプログラムの最後にある分岐命令によって、CPU110を、ROM40内のプログラムを実行する状態に戻す。そのとき、ROM40からのデータを入力とするスリーステートバッファ6が信号通過状態になっているので、問題なくROM40内のプログラム実行が再開される。

【0032】次に、図3のタイミング図を参照して、JMP命令コードROM120からのJMP命令コードの出力が完了する前にハードウェアによる割込みが発生した場合について説明する。レジスタ44には、特定アドレスデータとして「PCI+2」が設定されているとする。図3(m)に示すように、サイクル#1で割込み要求が発生したとする。このとき、CPU110は命令実行中であるから、割込み要求は、その命令の実行後に受け付けられる。従って、マイクロコンピュータにおいて、図3におけるサイクル#1~#8間で、割込み要求が発生しなかった場合と同様の処理が行われる。すなわち、図3におけるサイクル#1~#8でのマイクロコンピュータの処理は、図2におけるサイクル#1~#8での処理と同じである。なお、この場合のマイクロコンピュータの各部分の処理を示す図3(a)~(k)は、それぞれ、図2(a)~(k)に対応する。サイクル#8の終了時にCPU110のプログラムカウンタは「PCI+2」を示す(図3(d)参照)。

【0033】キューバッファ2は、サイクル#2~#4において、命令の先取り処理を行うためにアドレスバス

48にアドレス「PC1+2」を出力する(図3(g)参照)。よって、比較回路46は、一致信号をHレベルにする。すると、ラッチ回路14はセット状態になり出力がHレベルになるので、OR回路15は、ROMリード信号66を通過させない状態になる。すなわち、ROM40からデータバス50へのデータ出力は禁止される。同時に、AND回路16がROMリード信号66を通過させる状態になる。従って、HレベルのROMリード信号66に応じてデータjmp1, jmp2がJMP命令コードROM120からデータバス50に出力される。キューバッファ2は、データバス50からデータjmp1, jmp2を取り込む(図3(e)のサイクル#3, #4参照)。キューバッファ2内のデータ数は「2」になるので、キューバッファ2は、命令の先取り処理を中断する。よって、サイクル#8において、キューバッファ2には、JMP命令コードの一部であるデータjmp1, jmp2が格納されている。また、この時点で、ラッチ回路14はセット状態である。

【0034】割込み要求が既に発生していたので、一命令の実行が完了したサイクル#8の終了時において、CPU110は割込みによる分岐を開始する。この分岐を実行するために数サイクルの時間がかかるが、この分岐における最後のサイクルで、CPU110は、分岐発生信号68をHレベルにする。(図3(k)参照)。分岐発生信号68は、OR回路13を介してラッチ回路14をリセット状態にする。従って、OR回路15は、ROMリード信号66を通過させる状態になる。すなわち、ROM40からデータバス50へのデータ出力が可能な状態になる。同時に、AND回路16がROMリード信号66を通過させない状態になる。また、分岐発生信号68によって、キューバッファ2はクリアされる。

【0035】割込み分岐後に実行される割込み処理プログラムは、ROM40に格納されている。上述した処理によって、割込み処理プログラムの実行開始前に、ROM40からデータバス50へのデータ出力が可能な状態に戻される。また、キューバッファ2はクリアされているので、CPU110が割込み処理プログラムの先頭アドレスのデータをキューバッファ2に要求すると、キューバッファ2は、プログラムカウンタを処理プログラムの先頭アドレスに設定する。

【0036】仮に、ラッチ回路14を分岐発生信号68によってリセット状態にする上述した処理が行われなるとすると、ROM40からデータバス50へのデータ出力が禁止されJMP命令コードROM120からデータバス50へのデータ出力が許可された状態のままである。その場合には、JMP命令コードROM120が出力することになる命令jmp3, jmp4が、割込み分岐後の割込み処理プログラムの先頭でキューバッファ2を介してCPU110に読み込まれることになり、割込み処理プログラムを正常に実行することができない。し

かし、この実施の形態1においては、そのようなことはない。

【0037】割込み処理プログラムが終了すると、CPU110のプログラムカウンタは元の状態に戻される。すなわち、「PC1+2」になる。よって、CPU110はアドレスデータ62として「PC1+2」を要求する。割込み処理プログラムの終了時には一般に分岐命令が実行されるので、キューバッファ2はクリアされている。よって、アドレス「PC1+2」の要求に応じて、キューバッファ2は、プログラムカウンタを「PC1+2」に設定し、アドレスバスに「PC1+2」を出力する。すると、比較回路46がHレベルの一致信号を出力するので、再度、JMP命令コードROM120からデータバス50にJMP命令コードが出力される。よって、正しいプログラムを実行するための処理が行われる。

【0038】以上のように、この実施の形態1によれば、命令の先取り処理を行うキューバッファ2を有するマイクロコンピュータにおいて、キューバッファ2がJMP命令コードROM120からJMP命令コードを読み出しているときに割込みが発生しても、処理上何等の矛盾も生じない。すなわち、そのような場合には、JMP命令コードROM120からのJMP命令コードの読み出しが中断され、割込み処理プログラムの先頭アドレスの命令コードから順次命令コードがROM40からキューバッファ2に供給される。

【0039】なお、CPU110がプログラム中の分岐命令を実行しているときにキューバッファ2がアドレスバス48に「PC1+2」を出力することがあり得る。その場合、比較回路46がHレベルの一致信号を出力してラッチ回路14がセット状態にされるので、ROM40からデータバス50へのデータ出力が禁止される。そのままでは、ROM40からデータバス50へのデータ出力禁止状態が継続するので、分岐命令による飛び先のプログラムが実行できない。しかし、この実施の形態1によれば、分岐命令の実行完了時に分岐発生信号68によってラッチ回路14がリセットされるので、分岐命令による飛び先のプログラムが実行できなくなるといった事態は生じない。

【0040】実施の形態2. 図4はこの発明の実施の形態2によるマイクロコンピュータの構成を示すブロック図である。図において、122は図1におけるJMP命令コードROM120に代えて設けられたソフトウェア割込み命令コードROM(第2のROM)である。その他の構成要素は、図1に示された構成要素と同じものである。

【0041】次に動作について説明する。ソフトウェア割込み命令コードROM122には、特定番地に分岐する命令であるソフトウェア割込み命令があらかじめ固定されている。ソフトウェア割込み命令は、複数バイトで構成されているとする。比較回路46がHレベルの一致

信号を出力すると、実施の形態1の場合と同様に、ROM40からデータバス50へのデータ出力が禁止されるとともに、ソフトウェア割込み命令コードROM122からデータバス50へのデータ出力が許可される。従って、ROMリード信号66がHレベルになると、ソフトウェア割込み命令コードが、ソフトウェア割込み命令コードROM122からデータバス50に出力される。ソフトウェア割込み命令コードの出力が完了すると、制御回路8は、JMP命令コード終了信号70をHレベルにする。その他の動作は、実施の形態1における動作と同じである。

【0042】よって、この実施の形態2では、CPU110はJMP命令コードROM120に格納されたJMP命令を実行する代わりにソフトウェア割込み命令を実行するが、実施の形態1の場合と同様の効果を生じさせる。すなわち、正しいプログラムに分岐するためのソフトウェア割込み命令コードをソフトウェア割込み命令コードROM122からデータバス50に出力させたいときにのみ、ソフトウェア割込みコードROM122からのデータを入力とするスリーステートバッファ7が信号通過状態になり、スリーステートバッファ6が信号通過禁止状態になる。そして、CPU110が正しいプログラムに分岐するためのソフトウェア割込み命令を実行したら、ROM40からのデータを入力とするスリーステートバッファ6が信号通過状態になり、スリーステートバッファ7が信号通過禁止状態になる。

【0043】また、キューバッファ2がソフトウェア割込み命令コードROM122からソフトウェア割込み命令コードを読み出しているときに割込みが発生しても、処理上何等の矛盾も生じない。すなわち、そのような場合には、ソフトウェア割込み命令コードROM122からのソフトウェア割込み命令コードの読み出しが中断され、割込み処理プログラムの先頭アドレスの命令コードから順次命令コードがROM40からキューバッファ2に供給される。

【0044】実施の形態3。図5はこの発明の実施の形態3によるマイクロコンピュータの構成を示すブロック図である。ここでも、シングルチップマイクロコンピュータを例にする。図において、18はCPU110へのデータの入力元としてキューバッファ2とJMP命令コードROM120とのうちのいずれかを選択するスイッチ回路、61はROMリード信号66がHレベルになると信号通過状態になりROM40から出力されたデータをデータバス50に通過させるスリーステートバッファである。その他の構成要素は、図1に示された構成要素と同じものである。この場合には、比較回路46には、レジスタ44の出力とCPU110が出力するアドレスデータ62とが入力される。また、ラッチ回路14は、JMP命令コード終了信号70のみによってリセット状態にされる。ラッチ回路14の出力は、スリーステート

バッファ7および制御回路8に直接入力する。そして、スイッチ回路18は、ラッチ回路14の出力がLレベルのときにキューバッファ2の出力がCPU110に供給されるように切り換えられる。

【0045】図6は実施の形態3によるマイクロコンピュータの動作を説明するためのタイミング図である。この場合のマイクロコンピュータの各部分の処理を示す図6(a)～(k)は、それぞれ、図2(a)～(k)に対応する。ただし、図6(c)の括弧内は、キューバッファ2からCPU110に入力されたデータまたはJMP命令コードROM120からCPU110に入力されたデータを示す。

【0046】次に動作について説明する。レジスタ44には、「PCI+2」が設定されているとする。図6に示すサイクル#1に入る前では、キューバッファ2は初期状態であったとする。また、ラッチ回路14の出力はLレベルであって、CPU110は、キューバッファ2からのデータが入力する状態にあるとする。そして、CPU110は、キューバッファ2に対してアドレス「PCI」のデータを要求したとする(図6(d)のサイクル#1参照)。すると、キューバッファ2は「PCI」をアドレスバス48に出力するとともに、ROMリード信号66を有意なレベルであるHレベルにする。ラッチ回路14の出力はLレベルであるから、スリーステートバッファ7は信号通過禁止状態である。アドレスデコーダ4は、アドレスバス48上の「PCI」をデコードして、該当番地に格納されている命令コードをROM40に出力させる。スリーステートバッファ61はHレベルのROMリード信号66によって信号通過状態となるので、命令コードはデータバス50に出力される。キューバッファ2は、データバス50から命令コードを入力する。命令コードは2バイト単位で読み出されるので、ROM40は2バイトのデータop1, op2を出力し、キューバッファ2は、データop1, op2を入力する(図6(e)のサイクル#1, #2参照)。すなわち、キューバッファ2内のデータ数は「2」になる(図6(b)のサイクル#1, #2参照)。

【0047】CPU110からアドレス「PCI」のデータの要求があったので、キューバッファ2は、データop1をCPU110に対して出力する。よって、キューバッファ2内のデータ数は「1」になる(図6(b)のサイクル#2参照)。データ数が「1」になったので、キューバッファ2は、命令の先取り処理を実行する。すなわち、「PCI+2」をアドレスバス48に出力するとともに、ROMリード信号66をHレベルにする。一方、データop1が1つの命令コードのオペコードであることをCPU110が認識すると(図6(c)のサイクル#2参照)、キューバッファ2による命令の先取り処理の間、CPU110は、オペランドが格納されているアドレス「PCI+1」のデータをキューバッファ

10

20

30

40

50



ァ2に要求する(図6(d)のサイクル#2~#4参照)。キューバッファ2は、その要求に応じて、保持しているデータop2をCPU110に出力する。よって、キューバッファ2内のデータ数は「0」になる(図6(b)のサイクル#3参照)。CPU110は、サイクル#5~#8において、データop1, op2からなる命令に応じた処理を行う。

【0048】キューバッファ2が「PC1+2」をアドレスバス48に出力するとともにROMリード信号をHレベルにしたので、該当番地のデータop3, op4がROM40からデータバス50に出力される。キューバッファ2は、データバス50からデータop3, op4を入力する(図6(e)のサイクル#3, #4参照)。よって、キューバッファ2内のデータ数は「2」になる(図6(b)のサイクル#4参照)。データ数は「2」になったので、キューバッファ2は、命令先取り処理を中断する。なお、レジスタ44には「PC1+2」が設定されているが、比較回路46はアドレスバス48を導入していないので、この時点で、一致信号はHレベルにならない。

【0049】CPU110は、データop1, op2で構成される命令コードの処理が終了すると、キューバッファ2に、アドレス「PC1+1」の次のアドレス「PC1+2」の命令コードを要求する(図6(d)のサイクル#8, #9参照)。すると、比較回路46の一方の入力にも「PC1+2」が供給されるので、比較回路46は、Hレベルの一致信号を出力する(図6(h)参照)。従って、ラッチ回路14がセットされる(図6(j)のサイクル#9参照)。ラッチ回路14の出力がHレベルになるので、スリーステートバッファ7は信号通過状態になる。同時に、スイッチ回路18が、CPU110とスリーステートバッファ7とを接続するように切り換えられる。

【0050】制御回路8は、ラッチ回路14の出力がHレベルになると起動し、JMP命令コードROM120から、JMP命令を順序立てて出力させる。JMP命令コードROM120から、まず、命令コードの最初の1バイトであるデータjmp1が出力される。データjmp1は、スリーステートバッファ7およびスイッチ回路18を介してCPU110に入力する(図6(c)のサイクル#9参照)。

【0051】一方、キューバッファ2は、CPU110からアドレス「PC1+2」の命令コードを要求されたので、保持しているデータop3をCPU110側に出力する。しかし、スイッチ回路18はCPU110とキューバッファ2とを切り離しているため、このデータop3はCPU110には渡らない。キューバッファ2が1バイトのデータをCPU110側に出力したので、キューバッファ2内のデータ数は「1」になる(図6(b)のサイクル#9参照)。従って、キューバッファ2は、

命令の先取り処理を再開する。よって、アドレス「PC1+4, PC1+5」にあるデータop5, op6がROM40からデータバス50を介してキューバッファ2に入力する(図6(g)のサイクル#9~#11および図6(e)のサイクル#10, #11参照)。この結果、キューバッファ2内のデータ数は「3」になる(図6(b)のサイクル#10参照)。

【0052】CPU110は、データjmp1がJMP命令のオペコードであったことを知り、JMP命令のオペランドを得るために、キューバッファ2に「PC1+3」のデータを要求する(図6(d)のサイクル#9~#11)。キューバッファ2は、その要求に応じて保持しているデータop4, op5をCPU110側に出力する。スイッチ回路18はCPU110とキューバッファ2とを切り離しているため、データop4, op5はCPU110には渡らない。しかし、キューバッファ2内のデータ数は「1」になる(図6(b)のサイクル#11参照)。CPU110は、JMP命令コードの残りの1バイトを要求するために、アドレスデータ62として「PC1+5」を出力する。キューバッファ2は、この要求に応じて、保持しているデータop6をCPU110に転送する。スイッチ回路18はCPU110とキューバッファ2とを切り離しているため、データop6はCPU110には渡らない。しかし、キューバッファ2内のデータ数は「0」になる(図6(b)のサイクル#12参照)。

【0053】この間、CPU110には、スリーステートバッファ7およびスイッチ回路18を介してデータjmp2~jmp4が入力される。よって、CPU110のプログラムカウンタの値は、JMP命令のオペランドによる値になる。また、CPU110は、分岐発生信号68を出力する(図6(k)参照)。制御回路8は、JMP命令コードがJMP命令コードROM120から出力され終わった時点で、JMP命令コード終了信号70をHレベルに立ち上げる(図6(i)参照)。ラッチ回路14は、JMP命令コード終了信号70の立ち上がりによってリセット状態にされる。すなわち、その出力はLレベルになる(図6(j)のサイクル#14参照)。従って、スリーステートバッファ7は信号通過禁止状態になる。同時に、スイッチ回路18は、JMP命令コードROM120をCPU110から切り離し、キューバッファ2をCPU110に接続する。

【0054】キューバッファ2は、サイクル#11において、命令コードを先取りするために「PC1+6」をアドレスバス48に出力するとともにROMリード信号66をHレベルにする(図6(f), (g)のサイクル#11~#13参照)。HレベルのROMリード信号66に応じて、ROM40から、アドレス「PC1+6, PC1+7」のデータ「op, op8」がデータバス50に出力される。キューバッファ2は、データバス50からデ

ータ「op7, op8」を取り込む。しかし、キューバッファ2は分岐発生信号68によってクリアされるので、マイクロコンピュータは、実施の形態1の場合と同様に、JMP命令による飛び先にあるプログラムを実行する状態になる。

【0055】次に、図7のタイミング図を参照して、JMP命令コードROM120からのJMP命令コードの出力が完了する前に割込みが発生した場合について説明する。レジスタ44には、特定アドレスデータとして「PC1+2」が設定されているとする。図7(m)に示すように、サイクル#1で割込み要求が発生したとする。このとき、CPU110は命令実行中であるから、割込み要求は、その命令の実行後に受け付けられる。従って、マイクロコンピュータにおいて、図7におけるサイクル#1～#8間で、割込み要求が発生しなかった場合と同様の処理が行われる。すなわち、図7におけるサイクル#1～#8でのマイクロコンピュータの処理は、図6におけるサイクル#1～#8での処理と同じである。なお、この場合のマイクロコンピュータの各部分の処理を示す図7(a)～(k)は、それぞれ、図6

(a)～(k)に対応する。図6に示す場合と同様に、サイクル#8の終了時にCPU110のプログラムカウンタは「PC1+2」を示す(図7(d)参照)。

【0056】割込み要求が既に発生していたので、一命令の実行が完了したサイクル#8の終了時において、CPU110は割込み要求による分岐を開始する。プログラムカウンタの値である「PC1+2」は退避され、割込み処理プログラムの先頭番地の値がプログラムカウンタに設定される。従って、実際には、アドレスデータ62として「PC1+2」は出力されない。また、分岐発生信号68によって、キューバッファ2はクリアされる。割込み処理プログラムは、ROM40に格納されている。以後、キューバッファ2は、CPU110から要求された割込み処理プログラムにおける命令コードをROM40から入力し、CPU110に転送する。

【0057】この実施の形態3では、CPU110からのアドレスデータ62が比較回路46に直接入力されているので、アドレス「PC1+2」の直前の番地にある命令を実行しているときに割込みが発生しても、比較回路46からの一致信号はHレベルにならない。よって、ラッチ回路14の出力Hレベルにならない。従って、CPU110がJMP命令コードROM120に接続される状態にならないので、ROM40内の割込み処理プログラムは問題なくCPU110において実行される。

【0058】割込み処理プログラムが終了すると、CPU110のプログラムカウンタは元の状態に戻される。すなわち、「PC1+2」になる。よって、CPU110はアドレスデータ62として「PC1+2」を出力する。すると、比較回路46がHレベルの一致信号を出力するので、JMP命令コードROM120からCPU110

にJMP命令コードが出力される処理が開始される。

【0059】以上のように、この実施の形態3によれば、命令の先取り処理を行うキューバッファ2を有するマイクロコンピュータにおいて、JMP命令コードROM120からJMP命令コードが読み出される際に割込みが発生しても、処理上何等の矛盾も生じない。すなわち、そのような場合には、JMP命令コードROM120からのJMP命令コードの読み出しが待たされ、割込み処理プログラムの先頭アドレスの命令コードから順次命令コードがROM40からキューバッファ2に供給される。

【0060】上記の各実施の形態では、JMP命令コードROM120またはソフトウェア割込み命令コードROM122に分岐命令のみが格納されている場合について説明したが、必要に応じて他の命令も併せて格納するようにしてもよい。また、レジスタ44を複数のアドレス情報が格納できるようにし、複数のJMP命令コードROM120またはソフトウェア割込み命令コードROM122を設けてもよい。なお、レジスタ44のビット長は必ずしもアドレスバス48のビット長に一致している必要はない。ROMの全アドレス空間をアドレスバス48のビット長よりも短いビット長でアクセスできる場合には、レジスタ44のビット長はアドレスバス48のビット長よりも短くてよい。

【0061】上記の各実施の形態ではROM40が内蔵されたマイクロコンピュータを例にしたが、ROM40が外付けされるマイクロコンピュータであっても同様の効果を奏する。また、レジスタ44に不具合プログラムの先頭番地を設定するとともにRAM内の特定領域に正しいプログラムを書き込むためのEEPROMは、マイクロコンピュータに内蔵されていてもよい。

【0062】

【発明の効果】以上のように、請求項1記載の発明によれば、マイクロコンピュータを、分岐発生信号または命令コード終了信号を入力して第1のROMからのデータ出力を許可するとともに第2のROMからのデータ出力を禁止するROM出力切換部を備えるように構成したので、キューバッファを有する場合であっても、内蔵するROM内のプログラムの不具合を解消する機構を確実に動作させることができる効果がある。

【0063】請求項2記載の発明によれば、マイクロコンピュータを、アドレス情報格納部に格納されているアドレス情報とアドレスバス上のアドレスデータとを比較する比較回路からの一致信号でセット状態になり分岐発生信号または命令コード終了信号によってリセット状態になるラッチ回路とを含むように構成したので、一致信号、分岐発生信号および命令コード終了信号で制御されるラッチ回路の出力によって第1のROMおよび第2のROMのデータ出力状態が制御され、キューバッファを有する場合であっても、内蔵するROM内のプログラム

コンピュータの構成を示すブロック図である。

【図2】 実施の形態1によるマイクロコンピュータの動作を説明するためのタイミング図である。

【図3】 実施の形態1によるマイクロコンピュータの他の動作を説明するためのタイミング図である。

【図 4】 この発明の実施の形態 2 によるマイクロコンピュータの構成を示すブロック図である。

【図 5】 この発明の実施の形態 3 によるマイクロコンピュータの構成を示すブロック図である。

【図6】 実施の形態3によるマイクロコンピュータの動作を説明するためのタイミング図である。

【図 7】 実施の形態 3 によるマイクロコンピュータの他の動作を説明するためのタイミング図である。

【図8】 従来のシングルチップマイクロコンピュータの構成を示すブロック図である。

【図9】 ROM内のプログラムにおいて不具合が発見された状況に対処するための手法を説明するための図である。

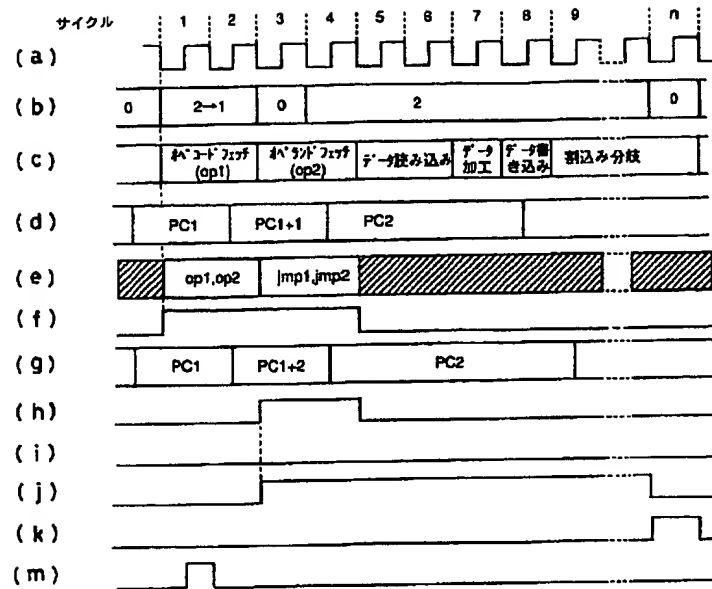
20 2 キューバッファ、6 スリーステートバッファ（ROM出力切換部、第2の論理回路）、7 スリーステートバッファ（ROM出力切換部、第4の論理回路）、14 ラッチ回路（ROM出力切換部）、15 論理和回路（ROM出力切換部、第1の論理回路）、16 論理積回路（ROM出力切換部、第3の論理回路）、18 スイッチ回路、40 ROM（第1のROM）、44 レジスタ（アドレス情報格納部）、46 比較回路（ROM出力切換部）、120 JMP命令コードROM（第2のROM）、122 ソフトウェア割込み命令コ

30 ードROM (第2のROM)。

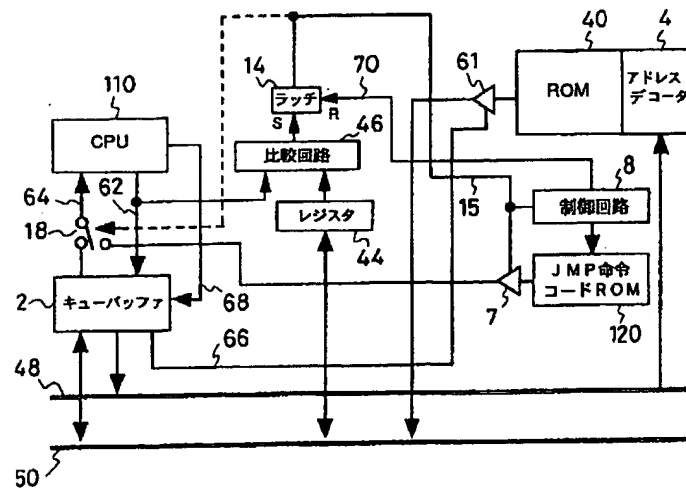
122: ソフトウェア割込み命令コードROM (第2のROM)



【図3】

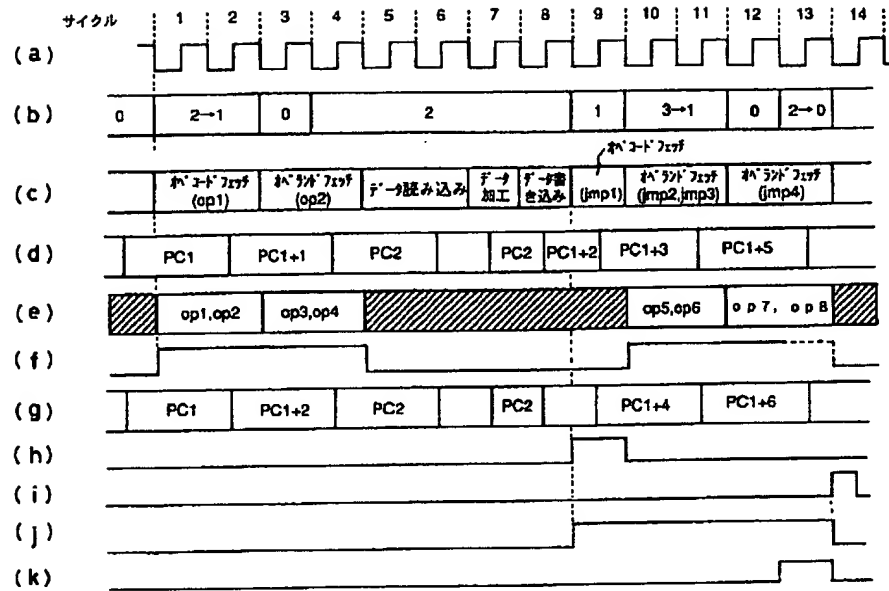


【図5】

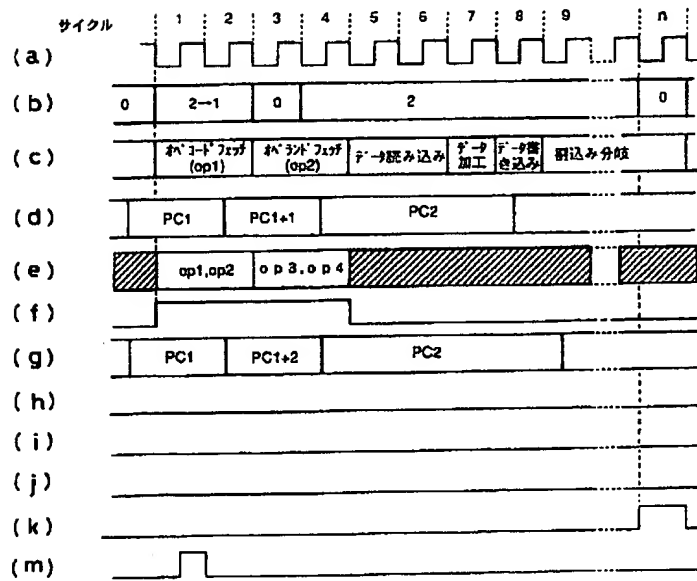


18: スイッチ回路

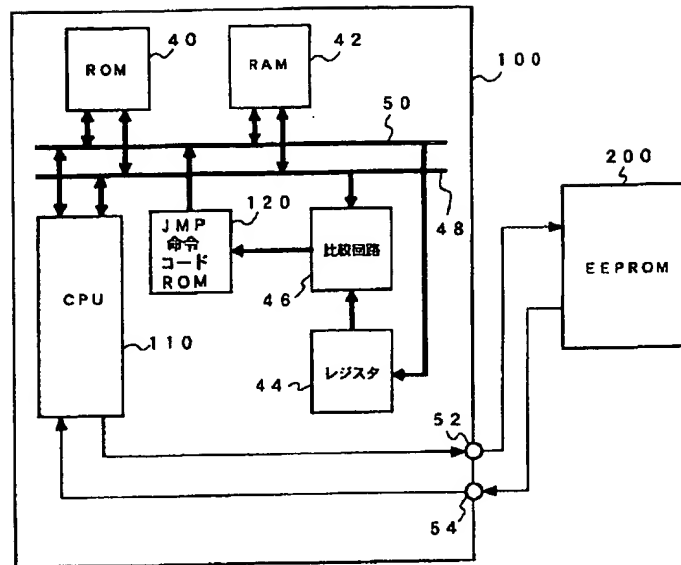
【図6】



【図7】



【図8】



【図9】

